DOCUMENT RESUME

ED 322 879 IR 014 486

AUTHOR	Epstein, Kenneth; Hillegeist, Eleanor
TITLE	The Role of Human Intelligence in Computer-Based Intelligent Tutoring Systems.
PUB DATE	17 Apr 90
NOTE	20p.; Paper presented at the Annual Meeting of the American Educational Research Association (Boston, MA, April 16-20, 1990).
PUB TYPE	Viewpoints (120) -- Reports - Descriptive (141) -- Speeches/Conference Papers (150)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	*Computer Assisted Instruction; Geometry; Higher Education; Learner Controlled Instruction; Microcomputers; *Programed Tutoring; Secondary Education; *Teacher Role
IDENTIFIERS	Apple Macintosh; *Geometry Proof Tutor; Intelligent Tutoring Systems

ABSTRACT

An Intelligent Tutoring System (ITS) consists of an expert problem-solving program in a subject domain, a tutoring model capable of remediation or primary instruction, and an assessment model that monitors student understanding. The Geometry Proof Tutor (GPT) is an ITS which was developed at Carnegie Mellon University and field tested in the Pittsburgh Public Schools. A version of the GPT for the Apple Macintosh microcomputer which was beta tested at Gallaudet University--and is still in use there in regular geometry classes--is a tutor rather than a primary instructor. It differs from the original GPT in that it does not contain a model of the domain of geometry and an expert geometry proof solver, and it restricts the students' explicit problem-solving to bottom up strategies. In this version, proofs developed by the original GPT are captured on a database and used by the MAC GPT in the form of expert solutions to which the system can refer. The following insights were gained during the test period: (1) since the teacher--rather than GPT--is responsible for teaching the fundamentals of geometry, the teacher's role is defined as providing necessary prerequisite knowledge; (2) the GPT provides effective guided practice for stronger students, which allows the teacher to work individually with those requiring more help; (3) teachers must facilitate the applications of new skills to different kinds of problems as the GPT cannot generalize outside of its own functions; and (4) although the GPT does not directly assess student performance, it has influenced the way teachers conduct student assessments, in particular, the level of detail now evident in student learning activities. While there remains a controversy over the role of humans in computer-assisted classrooms, it is concluded that the ITS will continue to offer teachers more freedom and creativity in their teaching as they begin to understand how they can subtly control the instructional environment to increase its effectiveness. (DB)

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

ED322879

The Role of Human Intelligence in Computer-Based Intelligent Tutoring Systems

Kenneth Epstein
Eleanor Hillegeist

Gallaudet University

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Kenneth Epstein
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Introduction

We believe that the key to successfully using computers in instruction is to clarify the roles of the students, the teacher, the computer, and the interactions among them. Our experience tells us that students and teachers must view the computer as a useful tool, not as either a threat or an impediment to good instruction. They must feel a sense of control and they must feel important in this new environment. Students and teachers need to be convinced through successful experiences that the computer is useful. No amount of "telling" a skeptical teacher or hesitant student is enough. A sense of control for the teacher develops from comfort with the software and hardware, knowledge of the content, freedom to structure the curriculum and activities, and assurance that he or she will remain part of the instructional system. Control for students includes freedom to make choices as they work with the computer and awareness that the instruction will be responsive to their needs. Importance follows from control and success. Students feel important when they discover that do have an element of control in shaping their learning. Their sense of importance grows as they begin to experience success and discover that they have developed skills and can solve

IR014494

BEST COPY AVAILABLE

problems that were originally beyond their abilities. Teacher's feelings of importance grow as they understand how they can subtly control the instructional environment to increase its effectiveness and as students begin to perceive them as valuable resources.

Based on our recent experience with an intelligent tutoring system (ITS), we are optimistic. We believe that teachers and computers can work well together. However, such cooperation does not come about automatically. Researchers, developers, and those of us who try to put technology to work must pay attention to the humans involved in instruction as well as the machines that support it. In this paper, we will explain the reason for our optimism and suggest some of the research and development activities that may will help achieve the expectations we have for computers in instruction.

Experience from educational technology is not encouraging. The use of systematically designed instruction during the last 30 years seems to have focused on decreasing the reliance on the teacher as a primary instructional resource. Often the teacher is described as a "manager" of instruction. However, with the exception of relatively short, self-contained, and restricted materials, most examples of instruction one finds in practice rely heavily on the human teacher for direct instruction.

Despite this persistent inability to find a way to eliminate the need for teachers, educational technologists do not appear to

have accepted the challenge of finding an appropriate role for them. We cannot find evidence to lead us to believe that much has changed since Convey (1978) pointed out in her discussion of multimedia instructional systems,

> "Little appears to be known about how human instructors can be used effectively in the attainment of desired cognitive and affective student outcomes. Educational technology, as it has been applied to instruction to date, appears to have allowed the instructor to "act on" the system (planning, implementing, evaluating, assessing) much more than it has helped him to "act within" the system (direct instructing, modeling, interacting). Predicted roles and actual roles for teachers have also, unfortunately, followed these same kinds of "acting on" activities." (p. 20)

At present, computer applications in education appear to be treated in much the same way as multimedia systems were in 1978. For example, Gagné, Wager & Rojas (1989) present guidelines for authors of computer-assisted instruction (CAI). Their article focuses on stand-alone CAI materials, with no suggestion that the responsibility for the instruction be shared with teachers. The situation is no better with respect to "intelligent" systems. Seidel and Park (1989), in their review of Etienne Wenger's book, Artificial Intelligence and Tutoring Systems, state, "The principle reason for the lack of widespread use of computers has not been the lack of intelligence in technology, but has, indeed, been a result of one of the points that Wenger mentions but does not emphasize, the lack of integrating the computer-aided instructional application into the larger context of educational institutions." (p. 382).

Intelligent Tutoring Systems

The term Intelligent Tutoring Systems was suggested by Sleeman and Brown (1982). They described the components necessary for a computer system to have a significant impact on instruction.

> "...the system must have its *own* problem-solving expertise, its own diagnostic or student modelling capabilities and its own explanatory capabilities. In order to orchestrate these reasoning capabilities it must also have explicit control or tutorial strategies specifying *when* to interrupt a student's problem-solving activity, *what* to say and *how* best to say it; all in order to provide the student with instructionally effective advice. Without such intelligent guidance, the student is liable to struggle uselessly with a conceptual roadblock that he is ill-equipped to handle, or to gloss over other situations that could have high instructional impact given his current state of knowledge. But by augmenting open-ended, problem solving environments with the above kind of tutorial intelligence, it becomes possible to transform a student's conceptual flounderings and misconceptions into profound and efficient learning experiences -- ones rooted in his own actions and hypotheses. In brief, the augmentation of environments with intelligent tutoring enables more students' misconceptions to be transformed into constructive learning experiences." (p. 2)

Wenger (1987) presents an expanded view of ITS. He sees the purpose of ITS research and development as building systems that

> "capture the very knowledge that allows experts to compose an instructional interaction...it is the knowledge itself that is explicitly represented so that it can be used in computer-based systems. It is then the responsibility of programs to compose instructional interactions dynamically, making decisions by reference to the knowledge with which they have been provided." (p. 5)

There have been a number of projects during the roughly 20 years of research with systems that include at least some of the characteristics of intelligent tutoring systems. There are

variations in approaches taken by each project, but the basic components of the ITS are similar. Clancey & Soloway (1990) list three essential components of any ITS. The system must include:

> "(1) an explicit model of the domain and an expert program that can solve problems in the domain,
>
> (2) a model of the student that identifies, at a fine-grained level of detail, what the student understands, and
>
> (3) a tutoring model that can provide instruction to remediate misconceptions and/or present new material." (p. 1)

Conceptually, the characteristics of intelligent tutoring systems seem very clear. One must represent the knowledge in a domain and build a program that the ITS can use to manipulate that knowledge to solve problems. Then, the ITS must be able to assess the knowledge and problem-solving skill of the learner and provide instruction that is most relevant to the individual. Building systems that are able to meet these requirements has proven to be very difficult, indeed.

The Geometry Proof Tutor (GPT) was developed in the early 1980's by John Anderson and his colleagues at Carnegie-Mellon University (Anderson, Boyle, and Yost, 1985; Anderson, Boyle, Corbett, and Lewis, 1990; Wertheimer, 1990). The original GPT was designed as a vehicle for testing Anderson's beliefs about the development of expertise. Structuring the knowledge of the domain of geometry and the rules for using that knowledge followed Anderson's ACT* theory. (ACT*, Adaptive Control of Thought is most thoroughly described in Anderson, 1983. A good summary

of the theory in its most recent form, PUPS, PenUltimate Production System, especially as it applies to the development of ITS, can be found in Anderson, Boyle, Corbett, and Lewis, 1990.) Originally, the system was used to generate geometric proofs. The research was concerned with determining whether it was possible to model the problem-solving behavior of human experts. Later, components representing the student model, tutorial strategies, and the interface were built to work with the knowledge and skill model. Results of field tests of the GPT with high school students in the Pittsburgh Public Schools were very encouraging, indicating that the GPT can help students learn to construct deductive geometric proofs.

In the summer of 1988, the Gallaudet University School of Preparatory Studies began to beta test a version of the GPT designed to operate on the Apple Macintosh computer. We are continuing to use it in regular geometry classes. The MAC version of the GPT is, to the best of our knowledge, the closest approximation to an ITS available commercially. We are hesitant to call it an Intelligent Tutoring System because of two important differences between the MAC version and the original GPT developed and tested in Pittsburgh. First of all, the MAC version does not contain a model of the domain of geometry and an expert geometry proof solver. Instead, proofs developed by the original GPT are captured on a database and used by the MAC GPT in the form of expert solutions to which the system can refer. Second,

the original GPT permitted both top down and bottom up reasoning. The MAC GPT restricts the students' explicit problem-solving to bottom up strategies. (Of course, students may be engaging in top down reasoning also, but they cannot demonstrate it explicitly in their interactions with the MAC GPT.) Despite these limitations, we feel that the MAC GPT provides an excellent illustration of the practical potential for ITS. It also permits us to explore the role of the teacher in an instructional environment enriched by this technology.

The Role of the Teacher

Everything necessary to teach oneself how to do the proofs included in the GPT curriculum is available within the software. One could conceivably work carefully through the tightly structured curriculum, asking for help and studying examples until the necessary rules and procedures for applying them were learned. However, this is not the manner in which the GPT is intended to be used.

The GPT is a tutor. It is not intended to be the primary instructor. The distinction is important because it clearly establishes different expectations and responsibilities for teachers, students, and the hardware and software.

The teacher using the GPT must assume the responsibility for teaching geometry. The software assumes that students do not need help learning the fundamentals of geometry. Thus, the teacher introduces and illustrates new ideas, explains the ra-

tionale for doing proofs, and teaches vocabulary and definitions. The GPT becomes important when the students are ready to practice applying their knowledge to solving proof problems. At that point, the tutor provides an opportunity to construct proofs that is, in a sense, supervised and supported by a knowledgeable expert.

One of us (Hillegeist) has been teaching with the GPT for several semesters. The changes she experienced as she learned how to use the GPT more effectively, provide insight into the changing role of the teacher in a technologically sophisticated classroom. We believe that our experiences can be generalized and that they will be relevant for others using ITS as well.

At first we used the GPT in a relatively unsystematic way. We simply asked students to try to do a selected number of proofs. At that point, we were learning the GPT ourselves, and we did not know if our students would be comfortable with the software. Our students are deaf and many of them have significant language problems, so we also wanted to ascertain whether the students could effectively deal with the level of language included in the help screens and rule review windows.

In our early experiments, we were not careful about teaching the pre-requisite geometry before allowing the students to work on the GPT. This often led to frustration on the part of the students. We have learned that the stronger the students' foun-

dation in the relevant geometry the more he or she will benefit from the GPT.

At first, we observed students using the GPT, but we were not active participants in the activities. We are outside observers no longer. Rather, the classroom has become a multifaceted learning environment, including the teacher, the students, and the GPT.

The Geometry Proof Tutor, as an example of an intelligent tutoring system, has a number of important characteristics which have led to a significant change in the way we approach instruction. The students tell us that they enjoy working with the GPT because

(1) it gives them the information to construct a proof when they need it,

(2) the feedback makes them feel important and encourages them to think, and

(3) they feel that they have control over their own learning.

In our opinion, the sense of control and the growing confidence in their abilities shown by the students are most important. While there are procedural rules and constraints, students feel that the GPT reacts to them. This is in marked contrast to most other instruction, including traditional CAI, in which the students have no control over the kind of instruction they receive or the practice they are expected to finish.

The environment that has been created feels dynamic. The program provides an opportunity to observe the way each student

attempts to solve each problem that is not present in a more traditional class. It creates a situation in which the teacher can be responsive to seizing a "moment," to teaching something that would, perhaps, have gone unnoticed. Working with the GPT has created opportunities for teaching that would probably not have been possible without it. All of the interactions, discussions, and diagnostic work would be much more cumbersome, and possibly impossible, in a more traditional environment. There is a feeling among the students that they can take charge of a significant portion of their learning when they use the GPT. Neither they nor their teacher is passive, all are actively involved in learning. This opens up special opportunities for teaching. The teacher can become more important to the students because the students are not waiting for the teacher to "deliver" knowledge. Instead, through their problem solving efforts, the students are identifying what they need to know and inviting their teacher to help them learn it.

In the case of an ITS that is explicitly intended to serve a tutoring function, such as the GPT, the obvious first responsibility of the teacher is to ensure that students have the prerequisite knowledge. Our suspicion is that, whether or not it is explicitly stated, any ITS that will be developed in the relatively near future will require that students have a solid knowledge base to begin using it. Thus, teachers will continue to have a critical role in preparing students to work with the ITS.

The GPT successfully provides guided practice for most students. In our experience, it has been most useful for relatively strong students. Weaker students tend to become frustrated or lost. The GPT does a good job of helping a good student "stay on track." It is less effective if the student has little or no idea of where to start. Therefore, we have found that we use a considerable amount of time helping the weaker students or the student who is having a problem with a particular proof. By providing support to the strong students, the GPT makes it possible for us to work with the students who need more help than the GPT can offer.

The GPT is an excellent tool to help us understand our students better. Watching a student working with the GPT provides a kind of window into his or her thinking processes that is difficult, if not impossible, to observe with more traditional instruction. For example, it helps identify and distinguish among students who are struggling with both factual and procedural problems by giving us the opportunity to observe the students in action. Our experience in the GPT classroom is also leading us to change our behavior in the regular classroom. It is giving us specific, immediate information about what students do well, where they struggle, and how we can improve our instruction.

The GPT knows about a limited number of geometry proofs. It has a particular way of approaching a problem and developing a proof. It does not know how to generalize. We can, and must,

help the students see how to apply their new skills to different kinds of problems. For example, we spend a considerable amount of time helping students transfer their problem solving skills to the two-column geometry proof format. We help students think about and verbalize the strategies they find useful in doing proofs. We help students develop planning strategies. We help students analyze their solutions, learn from their mistakes, and learn from other students' efforts. One unanticipated benefit of using the GPT is that students have begun spontaneously to ask each other for help and to offer to tutor each other.

The GPT does not directly support performance assessment. There is no internal record keeping, no explicit testing of skills, and no permanent history that could be used for assessment. It is true that it is technically possible to keep detailed student records, to the keystroke level if that is desirable. However, it is not clear to us how to make the best use of such data. There are many unanswered design questions concerning how to use an ITS to help with performance assessment directly.

In our classroom, the GPT is used in several assessment activities. We are beginning to develop observation techniques that help us record a students activities at an interpretable level of detail. We are experimenting with videotape and "after-the-fact" review of students working on proofs. The teacher conference with the student watching his or her performance on videotape is a very powerful instructional tool. We are designing

tests, for grading purposes, that include paper and pencil as well as GPT items. Our use of the GPT has broadened our way of thinking about assessment, but it is a very teacher intensive task.

Thoughts on the Future

The GPT, while providing much needed practical experience, is limited. It was developed relatively early in the history of ITS and, especially in the MAC version, it is extremely constrained by hardware. What is in store for the future?

We expect that ITS will become far more sophisticated and complex in the comparatively near future. We hope that future systems will provide more diagnostic information. For example, we would like a listing that identified and characterized our students' errors. We would also like to be able to summarize each student's history as an improving proof constructor. We agree with Anderson, et.al. (1990) that tutors need to be sensitive to each student's patterns of success and failure. For example, the feedback to a student who performed a particular procedure correctly several times before committing an error should be different than the feedback provided to a student who has not yet been successful. We are pondering the question of providing increased freedom to students to explore new ideas. While we feel that an opportunity to play with ideas, such as that provided by the Geometric Supposer, is very valuable, we are not sure where it belongs in the system. We believe that achieving

the proper balance between providing enough guidance to avoid frustrating students and providing so much guidance that it becomes frustrating and constraining will be very tricky. In fact, it is likely that the proper balance will vary from student to student.

There is a considerable amount of research activity exploring different approaches to building ITS. Hardware is becoming more sophisticated and capable of managing very complex systems. New ideas in hardware and software including parallel processing systems, neural networks, and associative memory techniques will open up possibilities that have not been explored. Object oriented programing presents interesting possibilities for developing data structures that may be very useful and new software developments that will permit databases and expert systems to work better together also suggest interesting possibilities. (For example, in a recent IEEE video conference on expert systems and databases (March 28, 1990), one of the panelists described systems in which productions embedded in the expert system directly manipulate the database. This kind of architecture may be useful in an ITS which builds a dynamic model of the student and changes it as a function of the student's responses.)

We are looking forward to the opportunities that ITS will open for teachers. We believe that ITS can free teachers and students from many of the routine tasks they are forced to face now and give them much more control over their teaching and

learning than is currently possible. This freedom will permit teachers to be more creative in their teaching and to expand their role dramatically beyond what it is in traditional classrooms.

Epilogue

A tacit assumption underlying ITS research and development is that it is theoretically possible to achieve its goals. This assumption implies that the information-processing model makes sense, and that computer intelligence is a meaningful concept. There are those who disagree.

Dreyfus and Dreyfus (1986a,b) have a long history of criticizing the artificial intelligence community. They propose a 5 stage model of the development of expertise which sets theoretical limits on what can be expected of ITS built according to the information-processing model. The Dreyfus and Dreyfus stages are (1) novice, (2) advanced beginner, (3) competent, (4) proficient, and (5) expert. They claim that the first three stages are rule governed and consistent with the information-processing model. However, in order to move to proficiency and expertise, one must move beyond rules. Intuitive, holistic problem-solving behavior characterizes the true expert. They argue that explicit rules are only necessary for dealing with new problems. As a novice gains experience, he or she generalizes and begins to recognize problems and their solutions holistically. In effect, an expert

knows "how" to solve a problem without necessarily knowing or being able to explain "what to do."

The Dreyfus and Dreyfus position suggests that there will always be a role for humans because the ITS hardware and software can never be more than competent. Without denying the importance of teaching the knowledge and rules underlying a domain, any ITS will be limited. For example, it will never be an intuitively sophisticated problem solver. As a tutor, no ITS will ever "know" the reason that a student is struggling with a particular problem and be able to present just the right idea to help him or her overcome it. Human experts will be necessary to intuitively grasp students' special problems and help them generalize their experiences so that they can develop true expertise. The GPT certainly requires this kind of help from the teacher. It will be fascinating to discover whether more sophisticated systems of the future will continue to need this kind help.

A somewhat different, but equally fascinating, question is posed by French (1990). His article addresses the nature of intelligence and argues that the Turing Test "provides a guarantee not of intelligence, but of culturally-oriented human intelligence." (p. 54) French does not claim that computers will not or can not be intelligent. However, he does say that they will not be human and that it will always be possible to distinguish between machine intelligence and human intelligence. French argues that the keys to determining whether a machine is intel-

ligent or not will lie in determining whether it has abilities to categorize, generalize, make analogies, learn, and so forth (p. 65). Given an intelligent machine, its non-human nature will present interesting problems and possibilities for the humans who interact with it.

Roles for teachers using Intelligent Tutoring Systems

1. Ensure that students have the prerequisite factual elements, vocabulary, and skills.

2. Provide the rationale for the upcoming learning activity and connect it to prior learning.

3. Assist weaker students or students having problems with a particular example. Provide support when the ITS expertise has been exhausted.

4. Assist with activities that tend to support problem-solving, such as planning, introspection, and inspection of one's solution. Encourage students to practice these activities.

5. Provide performance assessment.

6. Help the students understand how their activities with the ITS are related to their improved reasoning abilities.

7. Develop open-ended, ill-defined problems.

8. Help the students connect their ITS activities to activities that do not involve the computer.

9. Help students generalize their knowledge and skills to other subject areas. Point out alternative problem-solving strategies and alternative solutions.

10. Generalize, ask researchable questions, and help bridge the gap between research and practice.

Anderson, J. R. (1983). The Architecture of Cognition. Harvard University Press: Cambridge, MA.

Anderson, J. R., Boyle, C. F., & Yost, G. (1985). The geometry proof tutor. Proceedings of the International Joint Conference on Artificial Intelligence, Los Angeles: CA.

Anderson, J. R., Boyle, C. F., Corbett, A. T., & Lewis, M. W. (1990). Cognitive modeling and intelligent tutoring. Artificial Intelligence, 42 (1), 7-49.

Clancey, W. J. & Soloway, E. (1990). Artificial intelligence and learning environments: Preface. Artificial Intelligence, 42 (1), 1-6.

Convey, R. A. (1978). Designing roles for the instructor in systematically designed multimedia instructional system. Unpublish-doctoral dissertation, Tallahassee, Florida: Florida State University.

Dreyfus, H. L. & Dreyfus, S. E. (1986a). Why skills cannot be represented by rules. In, N. E. Sharkey (ed.) Advances in Cognitive Science 1. New York: John Wiley & Sons. pp. 315-335.

Dreyfus, H. L. & Dreyfus, S. E. (1986b). Mind over Machine. New York: The Free Press.

French, R. M. (1990). Subcognition and the limits of the Turing Test. Mind, 99 (393), 53-65.

Gagné, R. M., Wager, W. & Rojas, A. (1989). Planning and authoring computer-assisted instruction lessons. In, R. M. Gagné (Ed.) Studies of Learning: 50 Years of Research. Tallahassee, FL: Learning Systems Institute. pp. 587-606.

Seidel, R. J. & Park, O. C. (1989). ITS: Still looking for an interdisciplinary and instructional focus. Journal of Educational Computing Research, 5 (3), 379-384.

Sleeman, D. & Brown, J. S. (1982). Intelligent Tutoring Systems. London: Academic Press.

Wenger, E. (1987). Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge. Los Altos, CA: Morgan Kaufmann Publishers, Inc.

Wertheimer, R. (1990). The geometry proof tutor: An "intelligent" computer-based tutor in the classroom. Mathematics Teacher, 83 (4), 308-317.